Mengenal Fungsi Waktu

Mungkin di antara Anda ada yang hanya mengetahui fungsi umum untuk menampilkan format waktu, seperti tahun, bulan, dan tanggal (Y-m-d) dan juga waktu (jam), seperti (H:i:s). Jika Anda mau belajar, masih banyak sekali fungsi-fungsi waktu yang dimiliki PHP yang harus kita ketahui, seperti Checkdate, Date, Getdate, dan lain sebagainya. Penulis akan mulai dari yang paling mendasar terlebih dahulu.

5.1 Fungsi Checkdate()

Checkdate(), digunakan untuk memeriksa keabsahan suatu bentuk tanggal, atau lebih tepatnya untuk validasi penanggalan. Fungsi ini akan menghasilkan nilai TRUE jika tanggal dinyatakan valid dan akan menghasilkan FALSE jika tanggal dinyatakan tidak valid.

Sintaks penggunaan fungsi ini dapat dilihat sebagai berikut.

```
Checkdate(bulan, hari, tahun)
```

Bulan memiliki parameter angka 1 hingga 12, yang mewakili penamaan bulan dari Januari hingga Desember.

Hari memiliki parameter angka dari 1 hingga 30/31, dan 28/29 jika itu adalah bulan Februari, yang mewakili hitungan hari dalam kurun waktu satu bulan.

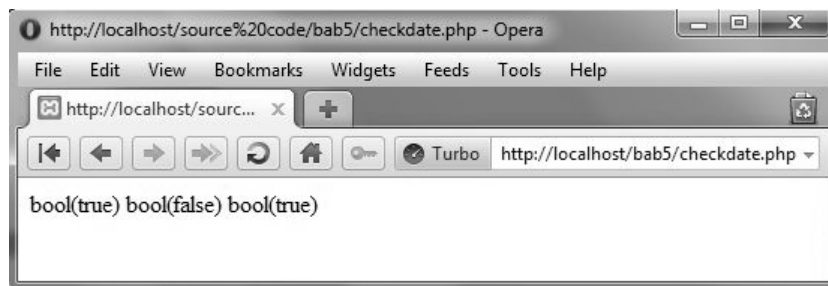
Tahun memiliki parameter angka valid dari 1 hingga 32767.

Untuk memperjelas teori seperti itu, kita akan hadirkan sebuah file praktek.. maksudnya, kita akan praktekkan.. gitu .. ☺

Sebagai persiapan, buat terlebih dahulu sebuah folder baru dengan nama folder **bab5** di dalam **htdocs** atau **www**.

Selanjutnya, buatlah file dengan nama **checkdate.php** dan simpan dalam folder **bab5** yang telah dibuat sebelumnya. Adapun skripnya sebagai berikut.

```
<?php
var_dump (checkdate(12,31,2000));
var_dump (checkdate(2,29,2003));
var_dump (checkdate(2,29,2004));
?>
```

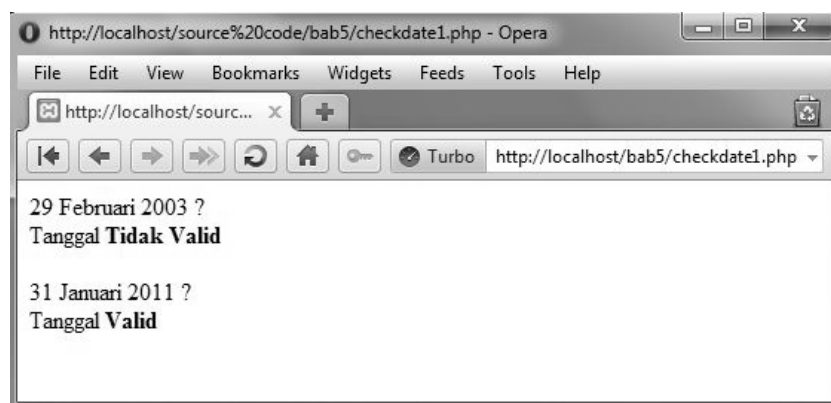
Nah, dari penulisan skrip tersebut, akan dihasilkan suatu nilai yang berisikan TRUE dan FALSE. Lihat Gambar 5.1.

Gambar 5.1 Checkdate 1

Terlihat sekali fungsi checkdate dapat melakukan keabsahan tanggal dengan sempurna. Hasil yang keluar FALSE, adalah tidak sah karena kita mengecek pada tanggal 29 Februari 2003 {checkdate(2,29,2003)}, yang notabene, tanggal tersebut tidak ada (bisa Anda cek pada kalender ☺).

Sebagai contoh yang lain, kita buatkan file lain dengan nama **checkdate1.php**, kemudian ketikkan skrip berikut.

```
<?php
$tes = checkdate(2,29,2003);
echo "29 Februari 2003 ? <br>";
if ($tes == TRUE) {
    echo "Tanggal <b> Valid </b>";
}
```

```
else {
    echo "Tanggal <b> Tidak Valid </b>";
}
?>
<br>
<br>
<?php
echo "31 Januari 2011 ? <br>";
$tes = checkdate(1,31,2011);
if ($tes == TRUE) {
    echo "Tanggal <b> Valid </b>";
}
else {
    echo "Tanggal <b> Tidak Valid </b>";
}
?>
```

Maka hasilnya dapat Anda lihat pada Gambar 5.2.

Gambar 5.2 Checkdate 2

5.2 Fungsi Date()

Fungsi Date, secara umum digunakan untuk menampilkan tanggal dan waktu saat ini sesuai format tanggal yang digunakan. Secara penulisan perintah, formatnya dapat dituliskan berikut.

```
date(‘Fungsi Tanggal’)
```

Misalnya:

```
date(‘Y-m-d’)
```

Simbol tersebut menunjukkan waktu Tahun-Bulan-Tanggal, sehingga penulisan skrip yang seperti itu akan menghasilkan format tanggal berikut.

2011-04-18

Untuk lebih jelasnya dapat Anda lihat daftar masing-masing karakter dan keterangan dari fungsi date berikut.

Format	Keterangan
a	"am" atau "pm"
A	"AM" atau "PM"
B	swatch internet time
d	day of the month, 2 digits with leading zeros ; ie. "01" to "31"
D	day of the week, textual, 3 letters ; i.e. "Fri"
F	month, textual , long ; i.e. "January"
g	hour, 12-hour format without leading zeros ; i.e. "1" to "12"
G	hour, 24-hour format without leading zeros; i.e. "0" to "23"
h	hour, 12-hour format ; i.e. "01" to "12"
H	hour,24-hour format ; i.e. "00" to "23"
i	minutes ; i.e. "00" to "59"
I (capital i)	"1" if Daylight savings time, "0" otherwise
j	day of the month without leading zeros ; i.e. "1" to "31"
l (lowecase L)	day of the week,textual,long ; i.e. "Friday"
L	boolean for whether it is a leap year ; i.e. "01" or "1"
m	month; i.e. "01" to "12"
M	month,textual,3 letters ; i.e. "Jan"
n	month without leading zeros ; i.e. "1" to "12"
s	seconds; i.e. "00" to "59"
S	English ordinal suffix,textual,2 charahcters; i.e. "th","nd"
t	number of days in the given month; i.e."28" to "31"
T	timezone setting in this machine;i.e. "MDT"
U	seconds since the epoch
w	day of the week, numeric, i.e. "0" (Sunday) to "6" (Saturday)
Y	year, 4 digits;i.e. "1999"
y	year, 2 digits; "99"
z	day of the year; i.e. "0" to "365"
Z	timezone offset in seconds (i.e. "-43200" to "43200")

Buat file dengan nama **date.php** kemudian ketikkan contoh skrip berikut.

```
<?php
$tgl = date('l, d F Y');
$jam = date('H:i:s A');
echo "Hari/Tanggal : ";
echo "<b> $tgl </b> <br>";
echo "Jam : ";
echo "<b> $jam </b>";
?>
```

Dan buka melalui web browser dengan mengetikkan url http://localhost/bab5/date.php, dan lihat hasilnya seperti ini.

Gambar 5.3 Fungsi Date

5.3 Date Default Timezone Set

Fungsi ini digunakan untuk menentukan atau menerapkan daerah waktu secara default yang akan digunakan oleh seluruh fungsi dalam sebuah dokumen PHP.

Untuk melihat daftar timezone yang didukung PHP, dapat dilihat pada situs resmi PHP, di http://www.php.net/manual/en/timezones.php.

Skrip dasar perintah Date Default Timezone Set, didasari oleh sintaks berikut:

```
Date_default_timezone_set('zona daerah')
```

Biasanya, untuk Indonesia menggunakan timezone "**Asia/Jakarta**" ataupun secara default (**UTC**).

Jadi, contoh pemakaiannya dapat Anda gunakan perintah berikut.

```
<?php
echo (date_default_timezone_set("Asia/Jakarta"));
?>
```

5.4 Fungsi Getdate()

Fungsi ini digunakan untuk mengambil atau mendapatkan informasi mengenai tanggal dan waktu saat ini (timestamp) serta berbentuk array.

Sintaks dasarnya sebagai berikut.

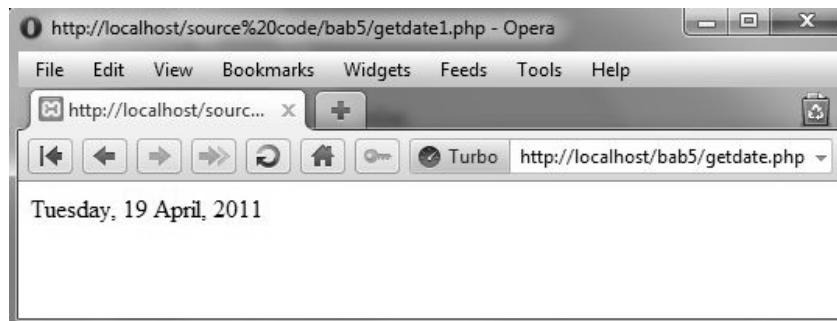
```
Getdate(timestamp)
```

Timestamp di sini memiliki beberapa format dan elemen, di antaranya yang bisa kita pergunakan:

- [hours] = jam
- [minutes] = menit
- [seconds] = detik
- [mday] = hari dalam satu bulan
- [wday] = hari dalam satu minggu
- [year] = tahun
- [yday] = hari dalam satu tahun
- [weekday] = nama hari dalam satu minggu, tekstual penuh, misalnya "Sunday"
- [month] = nama bulan dalam satu tahun, tekstual penuh, misalnya "January"

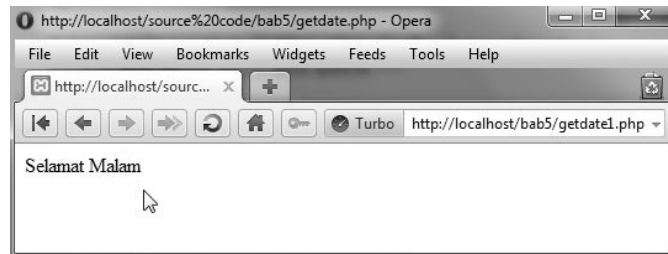
Jika tidak dipraktekkan, akan sangat sulit sekali untuk dijelaskan. Oleh sebab itu, buatlah file dengan nama **getdate.php** dan simpan dalam folder **bab5**, kemudian ketikkan skrip berikut.

```
<?php
$tgl = getdate();
print ("$tgl[weekday], $tgl[mday] $tgl[month], $tgl[year]");
?>
```

Setelahnya buka melalui web browser Anda dan ketikkan url http://localhost/bab5/getdate.php. Hasilnya akan terlihat seperti pada Gambar 5.4.

Gambar 5.4 Fungsi Getdate 1

Contoh lain dari penggunaan getdate, buat file dengan nama **getdate1.php** dan simpan dalam folder **bab5**, kemudian ketikkan skrip berikut.

```
<?php
$date = getdate();
$jam = $date['hours'];
if ($jam <= 11) {
    echo "Selamat Pagi";
}
else if ($jam > 11 and $jam <= 15) {
    echo "Selamat Siang";
}
else if ($jam > 15 and $jam <= 18) {
    echo "Selamat Sore";
}
else {
    echo "Selamat Malam";
}
?>
```

Buka web browser Anda dan ketikkan url http://localhost/bab5/getdate1.php. Hasilnya seperti Gambar 5.5.

Gambar 5.5 Fungsi Getdate 2

5.5 Fungsi idate()

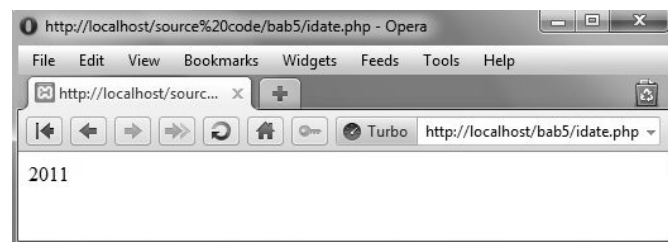
Fungsi **idate** digunakan untuk menset waktu dan tanggal lokal ke dalam bentuk *integer,* atau dalam bahasa manusianya "bilangan bulat".

Sintaks dasarnya sebagai berikut:

```
idate(format,timestamp)
```

Sebagai contoh, buatlah file dengan nama **idate.php** dan simpan dalam folder **bab5**, untuk kemudian ketikkan skrip berikut:

```
<?php
$idate = idate("Y");
echo "$idate";
?>
```

Buka hasilnya melalui web browser Anda, tentu saja dengan mengetikkan url http://localhost/bab5/idate.php. Akan didapatkan hasil seperti pada Gambar 5.6.

Gambar 5.6 Fungsi Idate

Seperti yang sudah dijelaskan bahwa fungsi ini hanya mampu menampilkan ke dalam bentuk bilangan bulat sehingga tanggal dalam format string tidak akan ditampilkan (error).

Berikut parameter yang dapat digunakan dalam fungsi **idate**.

Karakter	Keterangan
B	Swatch Beat/Internet Time
d	Day of the month
h	Hour (12 hour format)
H	Hour (24 hour format)
i	Minutes
I (uppercase i)	returns *1* if DST is activated, *0* otherwise
L (uppercase l)	returns *1* for leap year, *0* otherwise
m	Month number
s	Seconds
t	Days in current month
U	Seconds since the Unix Epoch - January 1 1970 00:00:00 UTC - this is the same as
w	Day of the week (*0* on Sunday)
W	ISO-8601 week number of year, weeks starting on Monday
y	Year (1 or 2 digits - check note below)
Y	Year (4 digits)
z	Day of the year
Z	Timezone offset in seconds

5.6 Fungsi Strtotime()

Fungsi ini digunakan untuk mengonversi english textual time ke dalam bentuk timestamp. Atau bahasa yang lebih mudah dicer-nanya, yaitu untuk mengubah tipe string menjadi format penang-galan/waktu.

Sintaks dasarnya sebagai berikut:

```
Strtotime(time, now)
```

Sebagai latihan, buatlah file dengan nama **strtotime1.php** dan simpan dalam folder **bab5**, kemudian ketikkan skrip berikut.

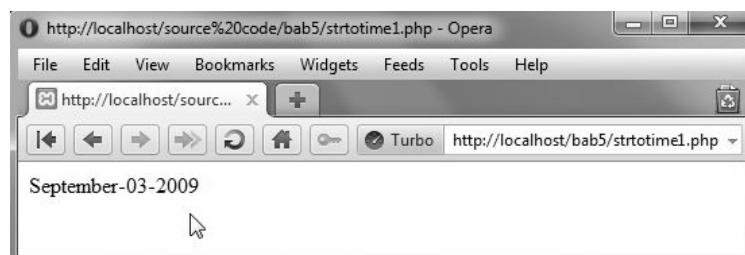
```
<?php
$date = date('F-d-Y', strtotime('3-9-2009'));
echo "$date";
?>
```

Dengan adanya perintah **strtotime**, penanggalan dapat dikonversi. Saat menulis buku, di Subbab 5.4 tercatat pada laptop (komputer) yang penulis gunakan adalah 20 April 2011, sehingga secara otomatis, seharusnya waktu adalah menunjuk saat ini, dan akan menghasilkan teks:

April-20-2011

Namun ternyata tidak, karena kita menggunakan strtotime, dan menset waktu menjadi 3-9-2011, maka akan ditampilkan hasil:

September-03-2009

Lihat Gambar 5.7.

Gambar 5.7 Fungsi Strtotime 1

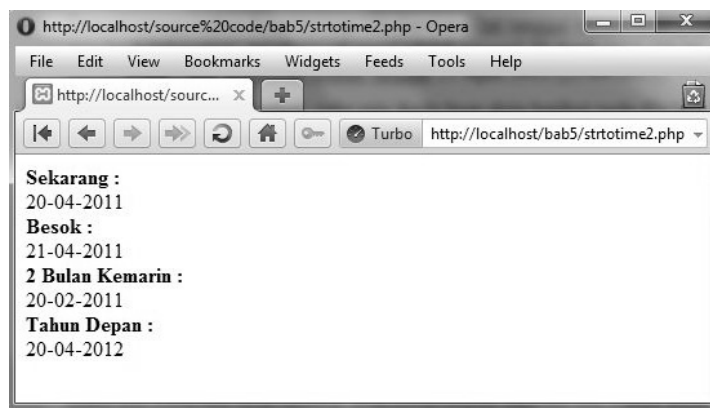
Dengan fungsi ini juga, kita dapat melihat tanggal berapa ke depan atau ke belakangnya. Misalnya sekarang adalah tanggal 20 April 2011 (20-04-2011), maka 5 hari kemudian adalah tanggal 25 April 2011 (25-04-2011).

Sungguh sangat mudah sekali, coba saja Anda buat skrip berikut pada file lain, misalnya Anda simpan dengan nama **strtotime2.php**.

```
<?php
$sekarang = date("d-m-Y");
$besok = date('d-m-Y', strtotime('+1 day'));
$duabulankemarin = date('d-m-Y', strtotime('-2 month'));
```

```
$tahun_depan = date('d-m-Y', strtotime('+1 year'));

echo "<b> Sekarang : </b> <br> $sekarang <br>";
echo "<b> Besok : </b> <br> $besok <br>";
echo "<b> 2 Bulan Kemarin : </b> <br> $duabulankemarin <br>";
echo "<b> Tahun Depan : </b> <br> $tahun_depan <br>";
?>
```

Dari kumpulan skrip tersebut akan menghasilkan halaman teks seperti pada Gambar 5.8.

Gambar 5.8 Fungsi Strtotime 2

5.7 Fungsi Time()

Fungsi **time()** digunakan untuk mendapatkan informasi saat ini. Secara format penulisan, sintaksnya dapat dituliskan sebagai berikut.

Time()

Hasil yang didapatkan dari fungsi ini adalah hitungan tanggal dan jam ataupun detik yang ditampilkan secara runtun.

Untuk memperjelasnya, dapat Anda buatkan file dengan nama **time.php**, kemudian simpan dalam folder **bab5**, setelahnya ketikkan skrip berikut.

```
<?php
$waktu = time();
echo "Waktu saat ini :<br> <b> $waktu </b>";
?>
```

Buka melalui web browser Anda, dan ketikkan url http://localhost/bab5/time.php. Hasilnya seperti pada Gambar 5.9.

Gambar 5.9 Fungsi Time()